A Novel Based Approach For Attendance Using Face Recognize System

Kaviyadharshini R[1], Kalaiyarasan K[2]
[1, 2]Dept of Computer Science And Engineering
[1, 2] Trichy Engineering College

Abstract- *In many of the educational institutions, managing attendance of students/candidates is tedious, as there would be large number of students in the class and keeping track of all is onerous. There are situations where student act as proxies for their friends even though they are not present. The advancement in the history of computer vision utilizing deep learning approaches especially convolutional neural networks have accomplished to solve difficult problems in face recognition field. Face recognition-based approach is one amongst the important identification methods which can be used as a possible substitution for conventional system of marking attendance manually, especially if a huge classroom of students is addressed for an hour session. Our solutions integrate AI capabilities with smart analytics features to facilitate transparency in classrooms and college campus.This project develops an automatic attendance system using Faster R-CNN deep learning based algorithm. In this system, a database containing the trained student's face. A camera installed in the college campus captures the face of all the student in the classroom and other places too. This face image is processed using FRCNN algorithms to detect faces and to mark the attendance automatically in an excel sheet. The system records the entire class session and identifies when the Students pay attention in the classroom, and then reports to the facilities and also this system can record violations of classroom, that is absence, roaming around the college campus during the class hours and send alert message to the H.O.D.This dynamic attendance system uses face recognition as an important aspect of taking attendance which saves time and proxy attendance and is avoided. The system identifies faces very fast needing only 100 milliseconds to one frame and obtaining a high accuracy. Our face recognition model has an accuracy rate of 99%.*

I. INTRODUCTION

Attendance is the act of being present at or attending a class, an event or a meeting. It depends upon the classroom, whether the attendance would be both physical and virtual. In schools,students must maintain their attendance against a set standard. The concept of students appearing in the classroom to attend the lecture is called class attendance. Class Teachers count all the students by their name and record the information in a register or digital software.

In recent years, **Artificial Intelligence (AI)** and particularly **Generative Adversarial**

all the students by their name and record the information in a register or digital software.

There are various methods of taking attendance like biometric systems using student identification cards or fingerprints, physical presence is recorded in registers or software tools. During online classes, the teacher marks online attendance to ensure the presence of all the students.

As per school policy, the attendance of students affects their grades. Sometimes, students' absenteeism from school has various reasons. Some of these reasons can be a bad company, personal issues, health issues, family problems, bullying, etc. Keeping a tab on absenteeism helps to identify the reasons behind a student's absence from school. When recognized, teachers can work in that direction to improve a student's situation. There are various methods of taking attendance like biometric systems using student identification cards or fingerprints, physical presence is recorded in registers or software tools. During online classes, the teacher marks online attendance to ensure the presence of all the students.

1.1 Deep Learning

Deep Learning is a subset of Machine Learning, which on the other hand is a subset of Artificial Intelligence. Artificial Intelligence is a general term that refers to techniques that enable computers to mimic human behaviour. Machine Learning represents a set of algorithms trained on data that make all of this possible.

1.2 Deep neural networks

A deep neural network is simply a shallow neural network with more than one hidden layer. Each neuron in the hidden layer is connected to many others. Each arrow has a weight property attached to it, which controls how much that

neuron's activation affects the others attached to it.The word 'deep' in deep learning is attributed to these deep hidden layers and derives its effectiveness from it. Selecting the number of hidden layers depends on the nature of the problem and the size of the data set. The following figure shows a deep neural network with two hidden layers.The data is fed into the input layer.

1.2.1. Types of neural networks

There are several types of neural networks.

CNN

The Convolutional Neural Networks or CNNs are primarily used for tasks related to computer vision or image processing.CNNs are extremely good in modelling spatial data such as 2D or 3D images and videos.

Faster – Region Convolutional Neural network
. It consists of 2 modules:

RPN: For generating region proposals.

Fast R-CNN: For detecting objects in the proposed regions.
The RPN module is responsible for generating region proposals. It applies the concept of attention in neural networks, so it guides the Fast R-CNN detection module to where to look for objects in the image.

1.3. OBJECTIVE

The aim of this project is to implement an advanced AI-based College Surveillance System incorporating Faster R-CNN and FaceNet technologies. The system aims to revolutionize attendance management, enhance campus security, and provide valuable behavioural insights in educational institutions. To automate attendance tracking for efficient and accurate record-keeping. To implement behavioral analysis for identifying patterns such as class skipping or unusual activities. To integrate FaceNet for precise face recognition and secure campus access.

To develop a Violation Alert module to identify and notify about attendance violations and unauthorized access. To create a comprehensive notification system for stakeholders' awareness and communication. To design a centralized College Admin Dashboard for efficient management of departments, semesters, and student information. To develop user interfaces for College Admin, Staff, and HoD roles with tailored functionalities. To implement the FaceNet model, involving video capture, preprocessing, face detection, feature extraction, CNN-based face recognition, and deployment with Faster R-CNN. To develop a Face Identification module for real-time face recognition in classrooms and campus areas. To automate attendance marking based on identified faces with real-time updates. To enhance user engagement through a user-friendly interface. To strengthen campus security with advanced technologies to detect and respond to potential threats.

1.4 SCOPE OF THE PROJECT

The project aims to implement an AI-based College Surveillance System, integrating Faster R-CNN and FaceNet technologies. It focuses on automating attendance management through precise facial recognition, implementing behavioral analysis for proactive intervention, and enhancing campus security with advanced features. The system includes a centralized administrative dashboard, tailored user interfaces, and a comprehensive notification system for efficient communication. Real-time face identification and automation of attendance processes are key objectives, contributing to a user-friendly interface and strengthening overall campus security.

II. LITERATURE SURVEY

2.1 Real-Time Classroom Monitoring Using Face Recognition and Computer Vision 2025

Maintaining discipline and monitoring student presence in classrooms is a major challenge in educational institutions. This work presents an AI-based classroom monitoring system using computer vision and deep learning techniques. The proposed system uses CCTV cameras to capture live video feeds and applies facial recognition algorithms to identify students and track their presence during lecture hours. The system can detect absenteeism and generate attendance reports automatically. Experimental analysis shows high recognition accuracy and reduced administrative workload. Nevertheless, this approach does not include behavior-based pattern analysis for detecting repeated class skippers.

2.2 AI-Driven Attendance and Activity Monitoring System for Smart Classrooms 2025

Smart classroom environments demand intelligent solutions for attendance and activity monitoring. This study presents a real-time AI-driven attendance system using Haar Cascade for face detection and LBPH for face recognition. The system captures classroom images and automatically updates attendance records. Experimental evaluation shows

improved accuracy and real-time performance. However, the system is limited to facial presence detection and does not analyze temporal patterns of student movement, which are necessary for identifying class skippers effectively.

2.3 AI-Based College Surveillance System for Detecting Class Skipping Behavior 2025

Class skipping is a recurring problem in higher education institutions, leading to poor academicperformance and reduced engagement. This paper proposes an AI-based college surveillance system using machine learning and computer vision techniques to monitor student presence and movement across classroom sessions. The system integrates face recognition with motion tracking to identify students who frequently leave or remain absent during lecture hours. Experimental results show effective detection of irregular attendance patterns. The study highlights the need for hybrid models combining facial recognition and behavior analysis for reliable class skipper identification.

2.4 Continuous Attendance Monitoring Using Deep Learning and CCTV Surveillance 2025

Continuous monitoring of student attendance using CCTV cameras has gained attention as an effective solution to address proxy attendance and class skipping. This paper presents a deep learning-based surveillance framework that integrates CCTV feeds with convolutional neural networks for continuous face recognition. The system monitors student presence throughout the lecture duration rather than at a single time instance. Experimental results indicate high accuracy in identifying students even in crowded classroom environments. However, the system lacks an alert mechanism to notify faculty about repeated class-skipping behavior in real time.

III.SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

Manual attendance calling, self-reporting attendance systems (using tools like Google forms), video calling students, short quizzes or polls, questions and discussions by selecting random students, and timed assignments.

In the case of physical classrooms, biometric-based attendance monitoring systems are essentially based on face, fingerprint, and iris recognition technologies Facial recognition is a technology that is capable of recognizing a person based on their face. It employs machine learning algorithms which find, capture, store and analyses facial features in order to match them with images of individuals in a

pre-existing database.Early approaches mainly focused on extracting different types of hand-crafted features with domain experts in computer vision and training effective classifiers for detection with traditional machine learning algorithms. Such methods are limited in that they often require computer vision experts in crafting effective features, and each individual component is optimized separately, making the whole detection pipeline often sub-optimal. There are many existing FR methods that achieve a good performance .

Support Vector Machine (SVM)

Support Vector Machines (SVM) are a popular training tool which can be used to generate a model based on several classes of data, and then distinguish between them. For the basic two-class classification problem, the goal of an SVM is to separate the two classes by a function induced from available examples. In the case of facial recognition, a class represents a unique face, and the SVM attempts to find what best separates the multiple feature vectors of one unique face from those of another unique face.

Principal Component Analysis (PCA)

One of the most used and cited statistical method is the Principal Component Analysis. A mathematical procedure performs a dimensionality reduction by extracting the principal component of multi-dimensional data.Principal component analysis id reducing the Eigen value and Eigen vectors problem in a matrix. Simply Principal component analysis is used for a wide range of variety in different applications such as Digital image processing, Computer vision and Pattern recognition.

Linear Discriminant Analysis (LDA)

LDA is widely used to find the linear combination of features while preserving class separability. Unlike PCA, the LDA tries to model to the difference between levels. For each level the LDA obtains differenced in multiple projection vectors.Linear discriminant analysis method is related to fisher discriminant analysis. Linear discriminant analysis is using to describing the local features of the images. Features are extracting the form of pixels in images; these features are known as shape feature, color feature and texture feature.

K-Nearest Neighbors

One of the basic classification algorithms in machine learning is known to be the k-NN algorithm. In machine learning, the k-NN algorithm is considered a well monitored type of learning. It is commonly used in the sorting of related

elements in searching apps. By constructing a vector representation of objects and then measuring them using appropriate distance metrics, the similarities between the items are determined.

3.2 PROPOSED SYSTEM

This paper proposed a system that can automatically attend using surveillance cameras. The camera will be installed in front of the class in an area that can reach all parts of the class and other area of the campus including vehicle stand, canteen, seminar hall, library etc. Propose a deep unified model for Face Recognition based on Faster Region Convolution Neural Network. Furthermore, by using a face detection system, the camera will mark the part of the frame from the captured image which is the face of students in a class and other area. The process is continued with an introduction to the face that has been detected so that it will automatically be kept a record that the student is present in the class or outside the class with a specific course and time.

FaceNet Model Integration

The FaceNet model is seamlessly integrated, involving video capture, frame preprocessing, face detection, feature extraction, CNN-based face recognition, and deployment with Faster R-CNN. This ensures high accuracy in face identification.

Real-time Face Identification Module

A dedicated module enables real-time face identification during live video feeds in classrooms and campus areas. This contributes to the overall automation of attendance processes, enhancing efficiency.

Face Recognition and Attendance Automation

The system utilizes FaceNet for precise face recognition, enabling automated attendance tracking. Real-time identification during class hours ensures accurate and efficient attendance management.

Behavioral Analysis with Faster R-CNN

Faster R-CNN is employed for behavioral analysis, detecting patterns like class skipping or unusual activities. This feature provides administrators with proactive insights into student engagement and potential challenges.

Violation Alert System

The system includes a Violation Alert module to detect unauthorized activities, such as students wandering during class hours. Administrators receive instant alerts with timestamped images, facilitating prompt intervention.

Comprehensive Notification System:

A robust notification system keeps stakeholders informed about attendance events, violations, and critical updates. Notifications, delivered through SMS, emails, or in-app messages, enhance communication within the educational community.

3.3. ALGORITHM

3.3.1 FRCNN

The Faster Region Convolutional Neural Network (Faster R-CNN) is a powerful object detection algorithm that combines deep learning with region proposal networks for efficient and accurate detection. Here is a step-by-step overview of the Faster R-CNN algorithm:

Input Frames:

The process begins with the input image, which is the target for object detection. The goal is to identify and locate objects within this image.

ConvolutionalBackbone (Feature Extraction):

The input image passes through a convolutional neural network (CNN) to extract features. This network, often pre-trained on a large dataset, captures hierarchical features at different scales.

Region Proposal Network (RPN):

The RPN operates on the feature map obtained from the convolutional backbone. It generates region proposals, which are potential bounding boxes around objects. These proposals are generated based on anchor boxes and their associated scores.

Anchor Boxes:

Anchor boxes are predefined bounding boxes of various scales and aspect ratios. The RPN suggests potential regions based on sliding these anchor boxes over the feature map.

Region of Interest (RoI) Pooling:

The proposed regions from the RPN are refined using RoI pooling. This step extracts fixed-size feature maps from each region proposal, allowing for consistent input sizes to the subsequent layers.

Fully Connected Layers:

The RoI-pooled feature maps are passed through fully connected layers to further refine and classify the proposed regions. These layers make predictions regarding the class of the object within each region and fine-tune the bounding box coordinates.

Bounding Box Regression:

Bounding box regression is applied to adjust the coordinates of the proposed bounding boxes. This step refines the localization of the detected objects.

Non-Maximum Suppression (NMS):

To eliminate redundant and overlapping detections, non-maximum suppression is employed. This process selects the most confident and accurate detections while discarding others.

Output:

The final output includes the detected objects along with their corresponding bounding boxes and confidence scores. This information provides a comprehensive understanding of the objects present in the input image.

Faster R-CNN's strength lies in its ability to integrate region proposal and object detection into a single unified framework. By leveraging deep learning for both feature extraction and region proposal, it achieves state-of-the-art performance in object detection tasks.

3.3.2. FaceNet Model

Input Frames:

The FaceNet model starts with a dataset of facial images. These images should be labeled with the identities of the individuals.

Face Detection:

Employ a face detection algorithm to locate and extract faces from the images. This step is crucial for isolating the facial features for subsequent processing.

Image Preprocessing:

Preprocess the extracted facial images, including resizing, normalization, and potentially alignment, to ensure standardized inputs for the model.

Triplets Sampling:

For training, FaceNet uses triplets of images for each example. A triplet consists of an anchor image (the reference face), a positive image (the same person as the anchor), and a negative image (a different person). Triplets are sampled to create a diverse training set.

Embedding Network (Siamese Network):

The core of FaceNet is the embedding network, typically implemented as a Siamese network. This network takes the anchor, positive, and negative images and maps them into a multi-dimensional feature space.

Triplet Loss Function:

The model is trained using a triplet loss function. This loss function encourages the embedding's of anchor and positive images to be close in the feature space, while pushing the negative image's embedding farther away. This process ensures that the model learns to distinguish between faces.

Training:

Train the model on a large dataset of triplets, optimizing the parameters to minimize the triplet loss. The goal is to obtain embedding's that accurately represent the similarities and differences between faces.

Embedding Extraction:

Once trained, the model can be used to extract embedding's (vectors) for new facial images. This embedding's are meaningful representations in a multi-dimensional space that capture facial features.

Face Recognition:

During the recognition phase, embedding's from different faces are compared in the feature space. A threshold is applied to determine whether two faces are considered a match or not.

Output:

The final output consists of facial embedding's, providing a robust representation of facial features suitable for tasks like face recognition, verification, and clustering.

IV. SOFTWARE DESCRIPTION

Python 3.7.4

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It was created by Guido van Rossum during 1985- 1990.
Python source code is also available under the GNU General Public License (GPL). This tutorial gives enough understanding on Python programming language.

Tensor Flow

TensorFlow is an end-to-end open-source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries, and community resources that lets researchers push the state-of-the-art in ML, and gives developers the ability to easily build and deploy ML-powered applications.

Keras

Keras is a deep learning API written in Python, running on top of the machine learning platform TensorFlow. It was developed with a focus on enabling fast experimentation.

Pandas

Pandas are a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language. Pandas are a Python package that provides fast, flexible, and expressive data structures designed to make working with "relational" or "labeled" data both easy and intuitive.

NumPy

NumPy, which stands for Numerical Python, is a library consisting of multidimensional array objects and a collection of routines for processing those arrays. Using NumPy, mathematical and logical operations on arrays can be performed.NumPy is a general-purpose array-processing package.

Matplotlib

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Matplotlib makes easy things easy and hard things possible.Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy.

Scikit Learn

Scikit-learn is a Python module for machine learning built on top of SciPy and is distributed under the 3-Clause BSD license. Scikit-learn (formerly scikits. learn and also known as sklearn) is a free software machine learning library for the Python programming language.

Pillow

Pillow is the friendly PIL fork by Alex Clark and Contributors. PIL is the Python Imaging Library by Fredrik Lundh and Contributors.

MySQL

MySQL is currently the most popular database management system software used for managing the relational database. It is open-source database software, which is supported by Oracle Company.

WampServer

]
WampServer is a Windows web development environment. It allows you to create web applications with Apache2, PHP and a MySQL database. Alongside, PhpMyAdmin allows you to manage easily your database..

FLASK

Flask is a web framework. This means flask provides you with tools, libraries and technologies that allow you to build a web application.

V. SYSTEM ARCHITECTURE

5.1 SYSTEM FLOW

Video Capture
Cameras installed across the college campus continuously capture video footage.

Frame Extraction
Video footage is processed to extract individual frames, creating a sequence of images.

Face Detection
Each frame undergoes face detection using a pre-trained model, isolating detected faces.

Image Preprocessing
Preprocessing steps enhance the quality of detected faces, including resizing, normalization, and denoising.

Face Recognition (FaceNet)
Preprocessed faces are passed through the FaceNet model for feature extraction and embedding.

Attendance Automation
The system compares obtained face embeddings with the database to automatically mark attendance.

Violation Detection
Student behavior during class hours is monitored, and violations, such as roaming, are detected.

Alert Generation
Alerts are generated for detected violations, providing details and timestamped images.

Notification
Automated notifications are sent to stakeholders based on attendance data, violations, or specific events.

Data Storage
Processed data, attendance records, and surveillance footage are stored in a database.

Reporting and Analysis
Administrators access a dashboard to view attendance reports, violation details, and analytics.

User Interaction
Users interact with the system through a user-friendly dashboard or application for monitoring and management.

Deployment with College Dashboard
The trained FaceNet model is seamlessly integrated into the College Dashboard for practical use.

VI. IMPLEMENTATION

6.1 College Management Dashboard

In this module we developed web browser-based interface designed to be used by college admin, Teaching Staff and Student– untrained in data science – to predict and detect face with mask and no mask for attendance System. It is a cloud based educational ERP software that enables higher education institutions & colleges to manage online admission, student enrolment, students' attendance, online fees, grades, assignments, library books, etc. Hosted on cloud, this college ERP helps educators to streamline all the core activities with the latest technology stack such as biometrics, business intelligence tools & analytics dashboard that generates precise reports on college admission, enrolment, scholarship, previous academic record, domicile, fees, alerts, attendance and compliance management.

6.2 End User

6.2.1. College Admin Dashboard

Login Page
Secure login page for College Admin with username and password fields.

Admin Home Page
Centralized dashboard for College Admin to manage departments, semesters, syllabus, students, staff, login credentials, face enrollment, timetable, attendance viewing, report generation, and notification customization.

Department Management
Page for adding and managing departments with an organized list of existing departments.

Semester Management
Page for adding and managing semesters with options to edit or delete existing entries.

Syllabus Management
Form to manage and organize syllabus details, including an editable list of existing entries.

Student Management
Form for adding and managing student details, coupled with a list and options to edit or delete.

Staff Management
Form for adding and managing staff details with an organized list for easy editing or deletion.

Login Credentials
Interface for College Admin to provide login credentials for staff and HoD.

Enroll Student Face
Dedicated page for capturing and enrolling student faces for attendance.

Timetable Management

Functionality to manage and organize class timetables for different departments and semesters.

View Attendance

Page to select department, semester, and date for viewing attendance records in a tabular format.

Generate Report

Form to select parameters for generating attendance reports with downloadable options.

Customize Notification

Form to customize and send notifications with options for recipients and message content.

6.2.2 Staff Dashboard

Login Page

Secure login portal for staff members with appropriate credentials.

Staff Home Page

Dashboard for staff to view timetable, student attendance, class skipper details, and images of roaming students.

View Timetable

Page displaying the timetable for the staff member's classes.

View Attendance

Page to select department, semester, and date to view attendance records in a tabular format.

Class Skipper Details

Page displaying details of students who have skipped classes, including images and additional information.

View Image of Students Roaming

Page displaying images of students found roaming around the college during class hours.

6.2.3. HoD Dashboard

Login Page

Secure login portal for the Head of the Department (HoD) with proper credentials.

HoD Home Page

Dashboard for HoD to view timetable, student attendance, class skipper details, and images of roaming students.

View Timetable

Page displaying the timetable for the HoD's department.

View Attendance

Page to select department, semester, and date to view attendance records in a tabular format.

Class Skipper Details

Page displaying details of students who have skipped classes, including images and additional information.

View Image of Students Roaming

Page displaying images of students found roaming around the college during class hours.

Send SMS to Parents

Functionality allowing HoD to send SMS notifications to parents regarding their ward's attendance or behavior.

6.3. FaceNet Model: Build and Train

This structured approach outlines the steps for building and training a Face Recognition model, specifically using FaceNet. The process involves capturing, converting, preprocessing, detecting faces, extracting features, and utilizing CNN for recognition. Additionally, the integration of Faster R-CNN for model development and training is emphasized. Finally, deploying the model within the College Dashboard system ensures practical and effective utilization. Specific implementation details may vary based on the chosen technologies and requirements.

6.3.1 Capture Live Face

Face Image Acquisition-

This module is initial part of the system. Logitech C270 (3MP) is used for image acquisition.

Frame Extraction

Frames are extracted from video input. The video must be divided into sequence of images which are further processed. The speed at which a video must be divided into images depends on the implementation of individuals. From we can say that, mostly 20-30 frames are taken per second which are sent to the next phases.

6.3.2 Pre-processing

Face Image pre-processing are the steps taken to format images before they are used by model training and inference.

The steps to be taken are:

Read image
RGB to Grey Scale conversion
Resize image
Original size (360, 480, 3) — (width, height, no. RGB channels)
Resized (220, 220, 3)
Remove noise (Denoise)
smooth our image to remove unwanted noise. We do this using gaussian blur.
Binarization

Image binarization is the process of taking a grayscale image and converting it to black-and-white, essentially reducing the information contained within the image from 256 shades of grey to 2: black and white, a binary image.

6.4 Face Identification

6.4.1 Live Feed from Class and Campus

This module establishes a continuous monitoring system by providing real-time video feeds from classrooms and various areas within the campus. The live feed integration allows for constant surveillance, enabling the system to capture events and activities as they unfold. This real-time aspect enhances the overall responsiveness of the Face Identification module, providing administrators with up-to-the-minute insights into student attendance and campus activities.

6.4.2 Face Recognition and Identification:

Powered by FaceNet, this component is pivotal for accurate attendance tracking and security enhancement. As live video feeds are received, the Face Identification module utilizes advanced facial recognition algorithms to identify and authenticate individuals in real time. The model compares captured facial features with the pre-trained facial database, ensuring precise identification. This not only automates attendance processes with a high level of accuracy but also contributes to campus security measures by instantly recognizing and validating individuals within the monitored areas. The utilization of FaceNet technology ensures a robust and efficient face recognition process, enhancing the reliability of the overall College Surveillance System. By seamlessly integrating live video feeds with state-of-the-art facial recognition capabilities, this component plays a key role in creating a secure and technologically advanced environment within the educational institution.

6.5. Attendance Automation

The Attendance Automation module stands as a pivotal component within the College Surveillance System, revolutionizing the traditional method of attendance management. By harnessing the power of advanced facial recognition technology, this module automates the attendance marking process, eliminating the need for manual recording. Employing the highly precise facial recognition capabilities of FaceNet, the system ensures accurate identification of individuals. This immediate and continuous updating of attendance records not only enhances accuracy but also significantly reduces the administrative burden associated with conventional attendance tracking methods. Integration with the College Admin Dashboard allows for effortless access to attendance records, generating reports and facilitating data-driven decision-making. This integration enhances overall efficiency, saving valuable time for educators and administrative staff. The tamper-resistant automated system guarantees accurate and verifiable attendance records, each linked to the identified face.

6.6. Violation Alert

The Violation Alert module within the College Surveillance System plays a critical role in maintaining a secure and disciplined learning environment. This proactive component continuously monitors student behavior and attendance patterns during designated class hours. The module employs advanced behavioral analysis to identify potential violations, such as students wandering or engaging in unauthorized activities within monitored areas. Upon detecting such violations, the system triggers immediate alerts, ensuring swift intervention by educators or administrators. To enhance the reliability of these alerts, the module captures images during violation instances, providing visual verification for administrators to assess thesituation accurately.

6.7. Notification

The Notification module within the College Surveillance System automates communication by sending timely updates to stakeholders based on attendance data, detected violations, and specific events. With customizable settings for administrators, faculty, and parents, this module ensures personalized notifications, fostering efficient and transparent communication. Whether alerting about attendance-related activities or violations, the system delivers information through various channels, such as SMS, emails, or in-app messages.

6.8. Reports

The Reports module in the College Surveillance System serves as a vital tool for generating detailed insights into attendance, behavioral patterns, and violation instances. Administrators can access customizable reports through the centralized dashboard, enabling them to analyze attendance trends, gain deeper insights into student behavior during class hours, and document violation instances. This data-driven approach empowers administrators to make informed decisions, implement proactive interventions, and continually enhance the educational environment..

VII. APPENDIXCES

Packages

```
from flask import Flask, render_template, Response, redirect, request, session, abort, url_for
from camera import VideoCamera
import cv2
import shutil
import datetime
frimport PIL.Image
from PIL import Image
import imagehash
import mysql.connector
import time
import imutils
import argparse
import shutil
```

Database Connection

```
mydb = mysql.connector.connect(
host="localhost",
user="root",
password="",
charset="utf8",
database="class_skipper"
```

Login

```
def login():
msg=""
if request.method=='POST':
uname=request.form['uname']
pwd=request.form['pass']
cursor = mydb.cursor()
cursor.execute('SELECT * FROM admin WHERE username = %s AND password = %s', (uname, pwd))
account = cursor.fetchone()
if account:
session['username'] = account['username']
return redirect(url_for('admin'))
else:
# Account doesnt exist or username/password incorrect
msg = 'Incorrect username/password!'
return render_template('login.html',msg=msg)
```

Add Student

```
def add_student():
#import student
msg=""
mycursor = mydb.cursor()
mycursor.execute("SELECT * FROM department")
value1 = mycursor.fetchall()
if request.method=='POST':
name=request.form['name']
mobile=request.form['mobile']
email=request.form['email']
address=request.form['address']
aadhar=request.form['aadhar']
regno=request.form['regno']
dept=request.form['dept']
year=request.form['year']
parent_mob=request.form['parent_mob']
mycursor.execute("SELECT count(*) FROM register where regno=%s",(regno,))
cnt = mycursor.fetchone()[0]
if cnt==0:
mycursor.execute("SELECT max(id)+1 FROM register")
maxid = mycursor.fetchone()[0]
if maxid is None:
maxid=1
sql = "INSERT INTO register(id,name,mobile,email,address,aadhar,regno,dept,year,parent_mob) VALUES (%s, %s, %s, %s, %s, %s,%s, %s, %s, %s)"
val = (maxid,name,mobile,email,address,aadhar,regno,dept,year,parent_mob)
mycursor.execute(sql, val)
mydb.commit()
return redirect(url_for('add_photo',vid=maxid))
```

Training

```
image = cv2.imread('static/dataset/'+fname)
original = image.copy()
kmeans = kmeans_color_quantization(image, clusters=4)
# Convert to grayscale, Gaussian blur, adaptive threshold
gray = cv2.cvtColor(kmeans, cv2.COLOR_BGR2GRAY)
blur = cv2.GaussianBlur(gray, (3,3), 0)
thresh = cv2.adaptiveThreshold(blur,255,cv2.ADAPTIVE_THRESH_GAUSSIAN_C, cv2.THRESH_BINARY_INV,21,2)
# Draw largest enclosing circle onto a mask
mask = np.zeros(original.shape[:2], dtype=np.uint8)
cnts = cv2.findContours(thresh, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)
cnts = cnts[0] if len(cnts) == 2 else cnts[1]
cnts = sorted(cnts, key=cv2.contourArea, reverse=True)
for c in cnts:
```

```
((x, y), r) = cv2.minEnclosingCircle(c)
cv2.circle(image, (int(x), int(y)), int(r), (36, 255, 12), 2)
cv2.circle(mask, (int(x), int(y)), int(r), 255, -1)
break
# Bitwise-and for result
result = cv2.bitwise_and (original, original, mask=mask)
result[mask==0] = (0,0,0)
path_main = 'static/dataset'
for fname in os.listdir(path_main):
##RPN
img = cv2.imread('static/trained/g_'+fname)
gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
ret, thresh = cv2.threshold(gray,0,255,cv2.THRESH_BINARY_INV+cv2.THRESH_OTSU)
kernel = np.ones((3,3),np.uint8)
opening = cv2.morphologyEx(thresh,cv2.MORPH_OPEN,kernel, iterations = 2)
# sure background area
sure_bg = cv2.dilate(opening,kernel,iterations=3)
# Finding sure foreground area
dist_transform = cv2.distanceTransform(opening,cv2.DIST_L2,5)
ret, sure_fg = cv2.threshold(dist_transform,1.5*dist_transform.max(),255,0)
# Finding unknown region
sure_fg = np.uint8(sure_fg)
segment = cv2.subtract(sure_bg,sure_fg)
img = Image.fromarray(img)
segment = Image.fromarray(segment)
def kmeans_color_quantization(image, clusters=8, rounds=1):
h, w = image.shape[:2]
samples = np.zeros([h*w,3], dtype=np.float32)
count = 0
for x in range(h):
for y in range(w):
samples[count] = image[x][y]
count += 1
compactness, labels, centers = cv2.kmeans(samples,
clusters,
None,
(cv2.TERM_CRITERIA_EPS + cv2.TERM_CRITERIA_MAX_ITER, 10000, 0.0001),
rounds,
cv2.KMEANS_RANDOM_CENTERS)
centers = np.uint8(centers)
res = centers[labels.flatten()]
return res.reshape((image.shape))
#Feature Extraction-Local Binary Patterns  (LBP)
###FR
id = 0
```

```
recognizer = cv2.face.LBPHFaceRecognizer_create()
recognizer.read('trainer/trainer.yml')
cascadePath = "haarcascade_frontalface_default.xml"
faceCascade = cv2.CascadeClassifier(cascadePath);

font = cv2.FONT_HERSHEY_SIMPLEX
gray=cv2.cvtColor(image,cv2.COLOR_BGR2GRAY)
faces = faceCascade.detectMultiScale(
gray,
scaleFactor = 1.2,
minNeighbors = 5,
minSize = (int(self.minW), int(self.minH)),
#FR-CNN – Face Identification
def FRCNN():
C.use_horizontal_flips = bool(options.horizontal_flips)
C.use_vertical_flips = bool(options.vertical_flips)
C.rot_90 = bool(options.rot_90)
C.model_path = options.output_weight_path
model_path_regex = re.match("^(.+)(\.hdf5)$", C.model_path)
if model_path_regex.group(2) != '.hdf5':
print('Output weights must have .hdf5 filetype')
exit(1)
C.num_rois = int(options.num_rois)
if options.network == 'vgg':
C.network = 'vgg'
from keras_frcnn import vgg as nn
elif options.network == 'resnet50':
from keras_frcnn import resnet as nn
C.network = 'resnet50'
else:
print('Not a valid model')
raise ValueError
if options.input_weight_path:
C.base_net_weights = options.input_weight_path
else:
# set the path to weights based on backend and model
C.base_net_weights = nn.get_weight_path()
train_imgs, classes_count, class_mapping = get_data(options.train_path)
val_imgs, _, _ = get_data(options.train_path)
```

ACKNOWLEDGEMENT

REFERENCES

[1] S. Kavin and A. Adhiselvam, "AI-Based Classroom Monitoring System," *International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering*, 2025.

[2] C. Sharma, D. Shukla, J. Kumar, and A. Verma, "AI-Based Attendance Monitoring System," *IJIRMPS*, Vol. 11, Issue 3, 2023.

[3] V. Lambu and S. Rampure, "Real-Time Student Attendance System Using AI and Face Recognition," *IJRASET*, 2025.

[4] "AI-Driven Attendance Tracking with Haar Cascade and ResNet," *Procedia Computer Science*, Vol. 258, 2025.

[5] L. Xu *et al.*, "ARIC: Activity Recognition in Classroom Surveillance Images," *arXiv*, 2024.

[6] Y. Qian *et al.*, "Few-Shot Continual Learning for Activity Recognition in Classroom Surveillance," *arXiv*, 2024.

[7] "Student behavior recognition for interaction detection in the classroom environment," *Image and Vision Computing*, 2023.

[8] Jaykumar Patel, Savita Gandhi, et al., "Enhancing Classroom Attendance Systems with Face Recognition through CCTV using Deep Learning," *Procedia Computer Science* 2025.

[9] Dharmaraj K B, Dhanushree C N, et al., "Real-Time Student Face Recognition Attendance System Using AI," *IARJSET*, Vol. 12, Issue 5, 2025.

[10] Sharma, Chhaya et al., "AI-Based Attendance Monitoring System," *IJIRMPS*, Vol. 11, Issue 3, 2023.